CONTENT RETRIEVAL FROM SITES THAT USE SESSION IDENTIFIERS

BACKGROUND OF THE INVENTION

A. Field of the Invention

[0001] The present invention relates generally to content retrieval on the world wide web, and more particularly, to automated web crawling.

B. Description of the Related Art

[0002] The World Wide Web ("web") contains a vast amount of information. Search engines assist users in locating desired portions of this information by cataloging web pages. Typically, in response to a user's request, the search engine returns references to documents relevant to the request.

[0003] Search engines may base their determination of the user's interest on search terms (called a search query) entered by the user. The goal of the search engine is to identify links to high quality relevant results based on the search query. Typically, the search engine accomplishes this by matching the terms in the search query to a corpus of pre-stored web documents. Web documents that contain the user's search terms are considered "hits" and are returned to the user.

[0004] The corpus of pre-stored web documents may be stored by the search engine as an index of terms found in the web pages. Documents that are to be added to the index may be automatically located by a program, sometimes referred to as a "spider," that automatically traverses ("crawls") web documents

based on the uniform resource locators (URLs) contained in the web documents. Thus, for example, a spider program may, starting at a given web page, download the page, index the page, and gather all the URLs present in the page. The spider program may then repeat this process for the web pages referred to by the URLs. In this way, the spider program "crawls" the world wide web based on its link structure.

[0005] Some web sites track users as they download different pages on the web site. User tracking is useful for identifying user behavior, such as identifying purchasing behavior by tracking the user through various web site page requests on a shopping orientated web site.

[0006] Two methods are commonly used to track user behavior: use of cookies to maintain information and embedding session identifiers in the uniform resource locators (URLs) in the web pages presented to the user. An embedded session identifier, in particular, may include a string of random characters embedded in the URLs returned to the user. When the user selects one of the URLs, the embedded identifier is returned to the web server in the request for the web page. The identifier can then be used to track the web pages presented to the user.

[0007] Embedding session identifiers in a web page, although potentially useful to the web site owner, poses problems to automated web spiders. Because the spider identifies pages based on their URLs, embedding session identifiers in a URL can cause the underlying web page to appear to be different to the spider each time a new session identifier is embedded in the URL. This

can, in turn, cause the spider to repeatedly crawl the same web page, thus limiting the spiders ability to crawl all possible sites.

[0008] Thus, there is a need in the art to more effectively crawl web sites that embed session identifiers in URLs.

SUMMARY OF THE INVENTION

[0009] The present invention is directed to techniques for identifying and using session identifiers in web documents.

[0010] One aspect of the invention is directed to a method of crawling documents. The method includes extracting a set of uniform resource locators (URLs) from at least one document and analyzing the extracted set of URLs to determine those in the set of URLs that contain session identifiers. The method further includes generating a clean set of URLs from the extracted set of URLs using the session identifiers and determining when at least one second URL has already been crawled based, at least in part, on a comparison of the second URL to the clean set of URLs.

[0011] Another aspect of the invention is directed to a method for receiving a set of uniform resource locators (URLs). The method includes analyzing the set of URLs for sub-strings that are consistent with session identifiers. The method additionally includes further analyzing the set of URLs to identify those of the sub-strings as corresponding to session identifiers based on multiple occurrences of a sub-string in the set of URLs.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate an embodiment of the invention and, together with the description, explain the invention. In the drawings,

[0013] Fig. 1 is an exemplary diagram of a network in which systems and methods consistent with the principles of the invention may be implemented;

[0014] Fig. 2 is an exemplary diagram of a client or server device according to an implementation consistent with the principles of the invention;

[0015] Fig. 3 is an exemplary functional block diagram illustrating an implementation of the server software shown in Fig. 1;

[0016] Fig. 4 is a flow chart illustrating the use of session identifiers by a web server;

[0017] Fig. 5 is a diagram illustrating a series of exemplary URLs that include session identifiers;

[0018] Fig. 6 is a diagram illustrating operations consistent with the invention through which a spider component may process URLs contained in downloaded web documents; and

[0019] Fig. 7 is a flowchart illustrating operations for crawling the web consistent with aspects of the invention.

DETAILED DESCRIPTION

[0020] The following detailed description of the invention refers to the accompanying drawings. The detailed description does not limit the invention.

[0021] As described herein, session identifiers are identified for a web host. URLs that contain the session identifiers may have the session identifier removed to generate "clean" URLs. The clean versions of the URLs can be used to identify the content corresponding to the URL.

EXEMPLARY NETWORK OVERVIEW

[0022] Fig. 1 is an exemplary diagram of a network 100 in which systems and methods consistent with the principles of the invention may be implemented. Network 100 may include multiple clients 110 connected to one or more servers 120 via a network 140. Network 140 may include a local area network (LAN), a wide area network (WAN), a telephone network, such as the Public Switched Telephone Network (PSTN), an intranet, the Internet, or a combination of networks. Two clients 110 and a server 120 have been illustrated as connected to network 140 for simplicity. In practice, there may be more or fewer clients and servers. Also, in some instances, a client may perform the functions of a server and a server may perform the functions of a client.

[0023] Clients 110 may include client entities. An entity may be defined as a device, such as a wireless telephone, a personal computer, a personal digital assistant (PDA), a lap top, or another type of computation or communication device, a thread or process running on one of these devices, and/or an object executable by one of these device. Server 120 may include server entities that process, search, and/or maintain documents in a manner consistent with the

principles of the invention. Clients 110 and server 120 may connect to network 140 via wired, wireless, or optical connections.

[0024] Clients 110 may include client software such as browser software 115. Browser software 115 may include a web browser such as the existing Microsoft Internet Explorer or Netscape Navigator browsers. For example, when network 140 is the Internet, clients 110 may navigate the web via browsers 115.

[0025] Server 120 may operate as a web server and include appropriate web server software 125. In one implementation, web server software 125 may function as a search engine, such as a query-based web page search engine. In general, in response to client requests, search engine 125 may return sets of documents to clients 110. The documents may be returned to clients 110 as a web page containing a list of links to web pages that are relevant to the search query. This list of links may be ranked and displayed in an order based on the search engine's determination of relevance to the search query. Although server 120 is illustrated as a single entity, in practice, server 120 may be implemented as a number of server devices.

[0026] A document, as the term is used herein, is to be broadly interpreted to include any machine-readable and machine-storable work product. A document may be an email, a file, a combination of files, one or more files with embedded links to other files, a news group posting, a web advertisement, etc. In the context of the Internet, a common document is a Web page. Web pages often include content and may include embedded information (such as meta

information, hyperlinks, etc.) and/or embedded instructions (such as Javascript, etc.).

EXEMPLARY CLIENT/SERVER ARCHITECTURE

[0027] Fig. 2 is an exemplary diagram of a client 110 or server 120 according to an implementation consistent with the principles of the invention. Client/server 110/120 may include a bus 210, a processor 220, a main memory 230, a read only memory (ROM) 240, a storage device 250, one or more input devices 260, one or more output devices 270, and a communication interface 280. Bus 210 may include one or more conductors that permit communication among the components of client/server 110/120.

[0028] Processor 220 may include any type of conventional processor or microprocessor that interprets and executes instructions. Main memory 230 may include a random access memory (RAM) or another type of dynamic storage device that stores information and instructions for execution by processor 220. ROM 240 may include a conventional ROM device or another type of static storage device that stores static information and instructions for use by processor 220. Storage device 250 may include a magnetic and/or optical recording medium and its corresponding drive.

[0029] Input device(s) 260 may include one or more conventional mechanisms that permit a user to input information to client/server 110/120, such as a keyboard, a mouse, a pen, voice recognition and/or biometric mechanisms, etc. Output device(s) 270 may include one or more conventional mechanisms

that output information to the user, including a display, a printer, a speaker, etc. Communication interface 280 may include any transceiver-like mechanism that enables client 110 to communicate with other devices and/or systems. For example, communication interface 280 may include mechanisms for communicating with another device or system via a network, such as network 140.

[0030] The software instructions defining server software 125 and browser software 115 may be read into memory 230 from another computer-readable medium, such as data storage device 250, or from another device via communication interface 280. The software instructions contained in memory 230 causes processor 220 to perform processes that will be described later. Alternatively, hardwired circuitry may be used in place of or in combination with software instructions to implement processes consistent with the present invention. Thus, implementations consistent with the principles of the invention are not limited to any specific combination of hardware circuitry and software.

[0031] As mentioned, server software 125 may implement a search engine that, based on a user query, returns a list of links to documents that the server software 125 considers to be relevant to the search.

SERVER SOFTWARE 125

[0032] Fig. 3 is an exemplary functional block diagram illustrating an implementation of server software 125. Server software 125 may include a search component 305, a database component 310, and a spider component

315. In general, search component 305 may receive user search queries from clients 110, search database 310 based on the search queries, and return a list of links (e.g., URLs) of relevant documents to the client 110. The list of links may also include information that generally attempts to describe the contents of the web documents associated with the links. The list of links may be ordered based on ranking values, generated by search component 305, which rates the links based on relevance.

[0033] Database component 310 may store an index of the web documents that have been crawled by spider program 315. In one implementation, the index of the web documents may be an inverted index. Database component 310 may be updated as new web documents are crawled and added to database component 310. Database component 310 may be accessed by search component 305 when responding to user search queries.

[0034] Spider component 315 may crawl documents available through network 140. Spider component 315 may include content manager 320, URL manager 325, and fetch bots 330. In general, fetch bots 330 may download content referenced by URLs. The URLs that are to be downloaded may be given to fetch bots 330 by URL manager 325. URL manager 325 may keep track of the URLs that have been downloaded and what URLs are to be downloaded. URL manager 325 may generate a "fingerprint" of each URL it receives by, for example, applying a hash function to the URL. The fingerprint can be used to quickly identify if a later-received URL is identical to one previously downloaded. Content manager 320 may receive content downloaded by fetch bots 330 and

destined for database component 310. Content manager 320 may process the content to extract URLs. The content may be forwarded to database component 310. URLs may be forwarded from content manager 320 to URL manager 325 and possibly to database component 310. The operation of content manager 320, URL manager 325, and fetch bots 330 consistent with aspects of the invention will be described in more detail below.

[0035] Although search component 305, database component 310, and spider component 315 are illustrated in Fig. 3 as all being part of server software 125, one of ordinary skill in the art will recognize that these components could be implemented on separate computing devices or clusters of computing devices. Spider component 315 and search component 305, in particular, may be implemented independently of one another.

[0036] Before describing the operation of spider component 315 in greater detail, it may be helpful to describe the way in which session identifiers are commonly used. Fig. 4 is a flow chart illustrating the use of session identifiers by a web server. As mentioned, session identifiers may be used to track the user of a client 110 as the user interacts with a particular host web site.

[0037] A user may begin by browsing a web site using browser software 115 (act 401). The web site may assign a session identifier to the user (act 402). The session identifier may be, for example, a string of characters, such as any string of characters that does not directly reference content. A web site that uses session identifiers to track user actions is typically a web site that includes multiple different possible web pages to which the user may navigate. A web

shopping site, for instance, may contain hundreds or thousands of possible web pages to which the user may navigate. Typically, after accessing the main web page for the shopping site, the user will navigate to other web pages in the shopping site by selecting URLs on the main web page or from later served web pages. When the user accesses the main web page for the web site, the web site may initially redirect the user to an alternate version of the main web page in which a session identifier is assigned to the user. For each additional web page returned to the user, the web site may include the session identifier in the URLs that point to other web pages on the web site (act 403). When the user attempts to access one of these URLs, the web site uses the session identifier to identify the user with the user's previous site accesses.

[0038] Fig. 5 is a diagram illustrating a series of exemplary URLs that include session identifiers. A number of URLs 501-504 are illustrated in Fig. 5. In this example, URLs 501-504 are URLs embedded in a web page for "somecompany.com." Each URL includes a session identifier 510 ("12341234"). If the user selects URL 502, for instance, the user's browser program 115 may contact "somecompany.com" and request the web page "/12341234/otherpage.htm." The "somecompany.com" web site may use session identifier 510, which the web server uses to identify the user's session. The web server also returns the content of the web page, "otherpage.htm." The URLs within "otherpage.htm" that refer to other web pages at "somecompany.com" may also contain session identifier 510.

[0039] Although session identifiers were described above as a specially inserted strings of characters, in general, any portion of a URL that doesn't reference content can be potentially used as a session identifier. Content, in this sense, does not necessarily mean that the web documents need to be exactly the same to be considered as having the same content. For example, web documents that are the same but for different color schemes, different advertisement links, or different navigation links may still be considered as having the same content.

[0040] Fig. 6 is a diagram illustrating operations consistent with the invention through which spider component 315 may process URLs contained in downloaded web documents. In general, spider component 315 operates to remove session identifiers from URLs to thus generate a clean version of the URLs. The clean version of the URLs (or an identifier based on the clean version) may then be stored by URL manager 325.

[0041] A fetch bot 330 may begin by crawling (downloading) the content for a particular web document that is specified by a URL (act 601). The URL to crawl may be specified by URL manager 325. Content manager 320 may then extract the URLs from the web document (act 602). Content manager 320 may alternatively extract the URLs from multiple web documents associated with a single web host and handle the URLs from the multiple web documents as if they originated from a single web document.

[0042] Content manager 320 may analyze the set of extracted URLs for session identifiers (acts 603). Session identifiers may be identified by searching

for strings that are structured in a manner consistent with session identifiers and that repeat across multiple URLs in the set of extracted URLs. For example, content manager 320 may initially look for sub-strings in the URLs in which the sub-string is not part of the URL domain name, contains a certain minimum number of characters (e.g., eight or more), and appears to be made of random characters. The randomness of a sub-string can be based on a comparison with terms from, for example, a dictionary. Or, for example, randomness can be estimated by keeping track of the number of times the string alternates between digits (0-9) lowercase alpha-numeric (a-z), and uppercase alpha-numeric (A-Z). For instance the string 3uSS4A has 4 such alternations, indicating a relatively high degree of randomness. Of these initially identified substrings, content manager 320 may classify a sub-string as a session identifier if the sub-string appears in multiple URLs in the set. One of ordinary skill in the art will recognize that a number of possible classification techniques are known in the art and could be used to initially locate the sub-strings in the URLs that are candidates for being session identifiers. Additionally, when classifying a sub-string as a session identifier based on multiple occurrences of the sub-string on a web site, additional factors, such as the general directory structure of the web site, may be taken into consideration.

[0043] For any URLs that contain session identifiers, content manager 320 may extract the session identifier from the URL (acts 604 and 605). With the session identifier extracted from the URL, the URL is a "clean" URL. The clean URL may be sent to URL manager 325, and possibly also to database 310 (act

606). In some implementations, a fingerprint of the URL, instead of or in addition to the actual URL, may be stored by URL manager 325. By storing clean URLs and comparing URLs based on their clean versions, URL manager 325 can determine when a URL has been previously crawled, even when the URL is one that was first identified with a session identifier.

[0044] Content manager 320 may additionally transmit the downloaded web documents to database 310. The URLs in the web documents transmitted to database 310 can be either the original URL or the clean version of the URL.

[0045] For some sites, to access the site, a session identifier is required, although the particular session identifier value does not matter. For other sites, the session identifier is required and must be a "valid" session identifier. Often, for these sites, a valid session identifier is defined as one that was previously issued by the web site.

[0046] When crawling network 140, if the site that is to be accessed is one that does not require a specific session identifier, URL manager 325 may generate a random session identifier that is defined to include a set of characters consistent with session identifiers generated by the web site. Thus, for these sites, when sending a URL to fetch bots 330, URL manager 325 may generate a session identifier and include it in the URL that is sent to fetch bots 330. However, for the purpose of storing the URLs and matching URLs that are later received in response to crawling the web, URL manager 325 may use the clean versions of the URLs.

[0047] For sites that require valid session identifiers, URL manager 325 may store and fingerprint the clean version of the URLs but use the URLs originally returned from the site when crawling additional documents on the site. In this situation, URL manager 325 may thus store the clean version of the URL and the session identifier. When the URL is needed for downloading purposes, URL manager 325 may substitute the session identifier back in the URL. Alternatively, URL manager 325 may store the originally received version of the URL and use the clean version of the URL for fingerprinting the URL. More generally, this technique of storing the originally downloaded version of the URLs but using the clean version for fingerprinting the URL may used for all sites when crawling the web.

[0048] Fig. 7 is a flowchart illustrating operations for crawling the web consistent with aspects of the invention. URL manager 325 may begin by identifying candidate URLs to crawl from previously crawled documents (act 701). The candidate URLs may include any URLs that were previously extracted from downloaded web documents. Alternatively, candidate URLs could be externally input to URL manager 325.

[0049] URL manager 325 may determine if the candidate URLs have been previously crawled by comparing clean versions of the candidate URLs to clean URLs that were previously crawled (act 702). The comparison may be based on a fingerprint of the URLs.

[0050] Candidate URLs that URL manager 325 determines should be crawled may be transmitted to fetch bots 330 (act 703). The URLs may be given to fetch bots 330 with the original or synthesized session identifiers.

CONCLUSION

[0051] As discussed above, session identifiers can be automatically identified in documents based on, among other things, multiple occurrences of a session identifier in URLs from a web site.

[0052] It will be apparent to one of ordinary skill in the art that aspects of the invention, as described above, may be implemented in many different forms of software, firmware, and hardware in the implementations illustrated in the figures. The actual software code or specialized control hardware used to implement aspects consistent with the present invention is not limiting of the present invention. Thus, the operation and behavior of the aspects were described without reference to the specific software code--it being understood that a person of ordinary skill in the art would be able to design software and control hardware to implement the aspects based on the description herein.

[0053] The foregoing description of preferred embodiments of the present invention provides illustration and description, but is not intended to be exhaustive or to limit the invention to the precise form disclosed. Modifications and variations are possible in light of the above teachings or may be acquired from practice of the invention.

[0054] No element, act, or instruction used in the description of the present application should be construed as critical or essential to the invention unless explicitly described as such. Also, as used herein, the article "a" is intended to include one or more items. Where only one item is intended, the term "one" or similar language is used.